下载APP

即学即练｜基础算法思想篇：复习卡 & 算法题特训

2022-02-01 黄清昊

《算法实战高手课》 课程介绍 >

讲述：黄清昊
时长 01:05 大小 1.01M

你好，我是微扰君。

大年初一，祝你新春快乐！

今天是春节特别策划的第二期。昨天的第一期是关于基础数据结构篇的，如果你错过了可以点这里答题 & 复习。

领资料

这 4 篇文章中的题目都是我精选出来的，无论是对知识点的理解，还是为了准备面试都是必须要掌握的，建议你一定要全部手写练习。如果一遍搞不定，你可以点击对应章节要点卡，再复习一下文章，多看几遍，结合题目反复练习，一定会收获很多。

基础算法思想篇

今天我们来复习 & 练习专栏的第二章，**基础算法思想篇**。在这个模块里，我们一起学习了外部排序、二分、搜索、字符串匹配、拓扑排序、哈夫曼树这些内容。

必知必会力扣题

题目名 846. 一手顺子

题目链接：https://leetcode-cn.com/problems/hand-of-straights/

题解思路

题目名 475. 供暖器

题目链接：https://leetcode-cn.com/problems/heaters/

题解思路

题目名 剑指 Offer II 069. 山峰数组的顶部

题目链接：https://leetcode-cn.com/problems/B1IidL/

题解思路

题目名 162. 寻找峰值

题目链接：https://leetcode-cn.com/problems/find-peak-element/

题解思路

题目名 1044. 最长重复子串

题目链接：https://leetcode-cn.com/problems/longest-duplicate-substring/

题解思路

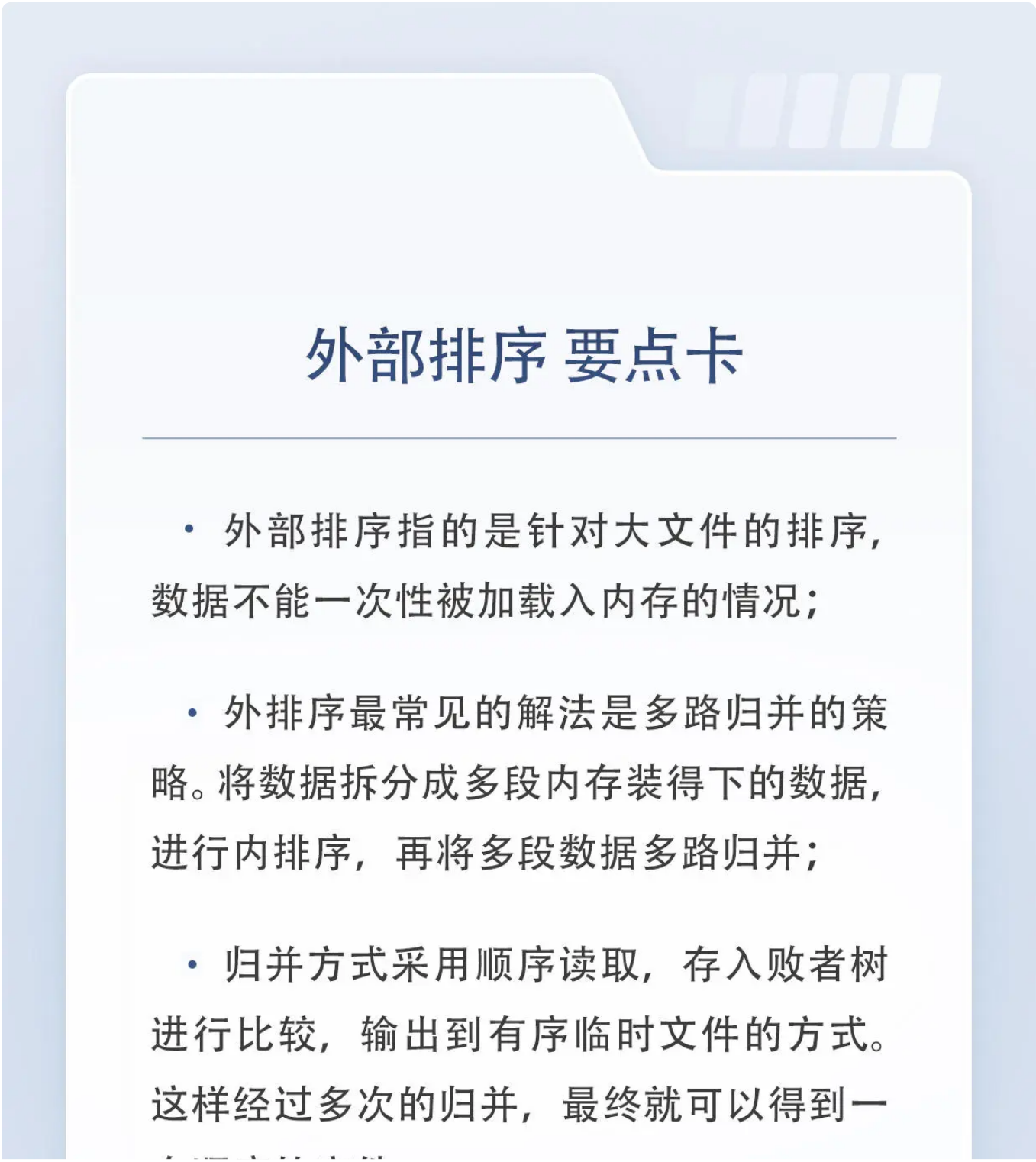
题目名 851. 喧闹和富有

题目链接：https://leetcode-cn.com/problems/loud-and-rich/

题解思路

复习要点卡

点击要点卡，直达你最需要复习的那篇。



个顺序的文件；

- 外排效率的提升主要在于两点：提高归并路数、尽量减少归并层数。但归并路数不能无限增加，会增加每次归并的时间复杂度。

二分 要点卡

- 二分是在有序序列上搜索目标元素的常见策略，搜索的时间复杂度为 O(logN)；
- 我们可以为无序的集合建立一次性的有序顺序索引，再利用二分搜索的特性，加速对无序集合元素的查询速度；

- 但有序索引通常只适用于不会改变或者 append only 的数据。因为数据一旦会动态变更，顺序索引会产生 O(N) 的删改时间复杂度，通常情况下是不可接受的；
- Kafka 就是通过建立离散顺序索引，加速消息查询的。

搜索 要点卡

- 搜索，是一种求解算法问题的常见方

式，最大特点就是尝试遍历搜索空间的每个情况，找到符合搜索目标的所有结果。

- 常见的搜索方式有两类：DFS 深度优先搜索、BFS 广度优先搜索；

- 深度优先搜索，通常采用递归的实现方式，比较容易在搜索过程中记录搜索路径；

- 广度优先搜索，需要借助于队列来实现，采用由内至外逐层推进地毯式的搜索策略。如果要求解一些诸如最短路径的问题，广度优先搜索的特性可以帮助我们大量减少搜索空间，效率更好。

字符串匹配 要点卡

- 字符串匹配，指的是对于长度为 N 的源字符串 s 和长度为 M 的模式串 p，查看 p 在 s 中是否有出现，并返回出现的位置。

- 常见做法包括时间复杂度为 O(N*M) 的 BF 算法、时间复杂度为 O(N)，利用预处理加速查询速度的 Boyer-Moore 和 KMP 算法、利用哈希和滑动窗口思想的 Rabin-Karp 算法，时间复杂度同样为 O(N)；

- Boyer-Moore 通过建立好后缀表和坏字符表，由后向前匹配，加速搜索过程，在 grep 中取得了非常好的效果。

拓扑排序 要点卡

- 拓扑排序解决的是在有向无环图中输出拓扑序度问题，通常对于一个图来说有多种情况；
- 拓扑序，描述节点间的顺序，序列中先出现的节点一定不会被后出现的节点所指向；
- 经典的两种思路分别基于 DFS 和 BFS 的思想，前者通常称为基于 DFS 的拓扑排序算法，后者我们通常称为 khan 算法。两者算法时间复杂度相当，都是 O(V+E)；
- 拓扑排序在工程中通常用于排序有依赖关系的任务。比如在编译过程中，可以用拓扑排序来决定依赖之间的编译顺序。

哈夫曼树 要点卡

- 哈夫曼树是经典的压缩方式，核心思想基于贪心算法，实质就是用更短的编码，来表示更常出现的字符。

- 哈夫曼树是一颗二叉树，每个叶子节点就代表着某个字符，左链代表 0，右链代表 1，从根节点到叶子节点的路径就代表着某个字符的编码方式。

- 节点的权重表示子树下所有字符出现的频率，每次选择最小的两个节点合并，我们就可以让更少出现的字符在树上更深的位置，更多出现的字符在更浅的位置

复习 & 练习的过程中，如果有什么困惑，获得了什么收获，都欢迎你留言讨论。做完题目之后，欢迎你把复习卡和你的 LeetCode 题解分享给你的朋友。

学习愉快，我们下期见～

分享给需要的人，Ta购买本课程，你将得 20 元

生成海报并分享

赞 1 提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。 页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 即学即练｜基础数据结构篇：复习卡 & 算法题特训

下一篇 即学即练｜操作系统篇：复习卡 & 算法题特训

学习推荐

JVM + NIO + Spring

各大厂面试题及知识点详解

限时免费

精选留言

写留言

> 由作者筛选后的优质留言将会公开显示，欢迎踊跃留言。